"use strict";

let blindSignatures = require("blind-signatures");

let SpyAgency = require("./spyAgency.js").SpyAgency;

// دالة لإنشاء مستندات بوثائق دبلوماسية

function makeDocument(coverName) {

return `The bearer of this signed document, ${coverName}, has full diplomatic immunity.`;

}

// دالة لتعمية المستند

function blind(msg, n, e) {

return blindSignatures.blind({

message: msg,

N: n,

E: e,

});

}

// دالة لفك التعمية عن التوقيع

function unblind(blindingFactor, sig, n) {

return blindSignatures.unblind({

signed: sig,

N: n,

r: blindingFactor,

});

}

// إنشاء كائن الوكالة التجسسية

let agency = new SpyAgency();

// تجهيز المستندات والهويات الوهمية

let documents = [];

let blindDocs = [];

let blindingFactors = [];

for (let i = 0; i < 10; i++) {

let coverName = `Agent ${i + 1}`;

let doc = makeDocument(coverName);

documents.push(doc);

let { blinded, r } = blind(doc, agency.n, agency.e);

blindDocs.push(blinded);

blindingFactors.push(r);

}

// توقيع المستندات من قبل الوكالة

agency.signDocument(blindDocs, (selected, verifyAndSign) => {

console.log(`Selected document index: ${selected}`);

// إعداد بيانات التحقق مع إزالة المستند المختار

let verifiedDocs = documents.map((doc, index) =>

index === selected ? undefined : doc

);

let verifiedFactors = blindingFactors.map((factor, index) =>

index === selected ? undefined : factor

);

// استدعاء التحقق والتوقيع

let blindedSignature = verifyAndSign(verifiedFactors, verifiedDocs);

// فك التعمية واستعادة التوقيع الحقيقي

let signature = unblind(

blindingFactors[selected],

blindedSignature,

agency.n

);

// طباعة النتائج المطلوبة

console.log(`Signed document: ${documents[selected]}`);

console.log(`Signature: ${signature}`);

});